International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Missing Child Identification Using DLIB And Gan Based Facial Augmentation

Karthi S [1],jayasaran S [2],NMK Ramalingam Sakthivelan[3],

1,2,3 Department of cse, Paavai Engineering college, Namakkal, India

.

ABSTRACT

This project presents a robust and intelligent system for identifying missing children using advanced facial recognition and object detection techniques. The proposed system is developed in three main stages. In the first stage, a dataset of children's faces is collected and trained using the Haar Cascade Classifier with the default frontal face XML, allowing for basic face detection and recognition. The second stage enhances accuracy by integrating the YOLOv8 object detection model, which is trained on the same facial dataset to provide faster and more reliable face identification in real-time environments. To increase detection efficiency, especially when the uploaded image quality is low or outdated, the third stage includes an image augmentation module. This module generates multiple variants of the input image through transformations like rotation, scaling, and brightness adjustment. These augmented images are added to the training set, enabling the model to recognize faces under varied conditions. When a match is successfully identified, an automated email alert is sent to the concerned individual or authority using Flask-Mail integration. This system leverages the power of artificial intelligence and automation to support faster and more accurate identification of missing children, significantly aiding efforts in child recovery and public safety.

INTRODUCTION

The issue of missing children remains a critical concern globally, demanding innovative technological solutions for faster and more accurate identification. Traditional manual methods of searching and identifying missing individuals are time-consuming and often yield limited results. To address this challenge, our project proposes an AI-based system that leverages facial recognition and deep learning techniques to assist in locating and identifying missing children efficiently. The system operates in three key stages.

The first stage involves collecting a dataset of facial images and using Haar Cascade Classifier with the default frontal face XML for basic face detection. This provides an initial level of identification. To improve accuracy and performance, the second stage employs the YOLOv8 (You Only Look Once version 8) object detection algorithm. YOLOv8 is trained specifically on facial datasets to enable real-time and precise face detection even in complex environments.

EXISTING SYSTEM

The existing systems for missing child identification largely rely on manual efforts, including posters, media announcements, and community alerts, which are time-consuming and often inefficient. Some advanced methods use facial recognition technology integrated with CCTV footage or government databases. These systems typically employ traditional face detection algorithms such as Haar Cascade and Eigenfaces. While they offer basic identification features, they lack accuracy in real-time environments and often fail to perform effectively under varied lighting, angle, or facial expressions. Additionally, current systems do not support dynamic learning or image augmentation, which limits their ability to detect changes in facial appearance over time. Notification systems, such as email or SMS alerts, are not commonly integrated into these platforms, reducing their effectiveness in timely communication. Thus, there is a need for an intelligent, real-time, and accurate system that uses advanced deep learning models and supports features like image augmentation and automated notifications.

Disadvantages of Existing System:

1. Low accuracy in real-time facial recognition, especially in crowded or poor lighting environments.
2. Cannot handle changes in appearance such as aging, hairstyle, or partial obstructions.
3. Lack of integration with automatic notification systems for immediate response.
4. No support for image augmentation to improve detection with limited data.
5. Slow processing and high chances of false positives or missed identifications.

PROPOSED SYSTEM

The proposed system aims to enhance the identification of missing children using an AI-powered facial recognition and object detection framework. It consists of three main stages: dataset collection and training using Haar Cascade for initial face detection, implementation of YOLOv8 for real-time and high-accuracy face recognition, and image augmentation for improved robustness. The YOLOv8 model is trained specifically on facial datasets to ensure accurate detection even in complex or dynamic environments. To address variations in appearance, augmented images (e.g., rotated, brightened, zoomed) of the uploaded photo are generated and used for better recognition. Once a match is found, the system triggers an automated email alert to the concerned individual using Flask-Mail. This integrated approach ensures a faster, more reliable, and intelligent solution for identifying missing children, supporting both law enforcement and families with real-time updates and enhanced accuracy.

Advantages of Proposed System:

1. High detection accuracy using YOLOv8 deep learning model.
2. Real-time face recognition with support for varied conditions and environments.
3. Image augmentation improves recognition performance from limited or altered images.
4. Automated email notifications ensure timely alerts to concerned authorities or guardians.
5. Scalable and adaptable system that can be integrated with CCTV and mobile platforms.

SYSTEM REQUIREMENTS

HARDWARE REQUIREMENTS:

- Webcam (for capturing real-time facial images)
- Computer/Laptop with minimum 4GB RAM and i5 processor (for training and detection tasks)
- Internet connection (for email notifications via Flask-Mail)

SOFTWARE REQUIREMENTS:

- Operating System: Windows/Linux
- Programming Language: Python 3.x

LIBRARIES & FRAMEWORKS:

- OpenCV (for image processing and Haar Cascade)
- YOLOv8 (Ultralytics package for object detection)

- Flask (for web-based interaction and backend)
- Flask-Mail (for sending email alerts)
- NumPy (for numerical operations)
- PIL (Python Imaging Library for image handling)

Annotation Tool: Roboflow or any tool for YOLOv8 dataset preparation

Model Format: YOLOv8 .pt file (trained model for detection)

HARDWARE REQUIREMENT:

Hardware Requirements: Detailed Explanation

The successful implementation of a facial recognition system for identifying missing children requires an efficient and robust hardware setup. Since this project involves real-time image capture, deep learning model execution, and internet-based communication, the hardware must be chosen to support these features without latency or system crashes. Below is a comprehensive explanation of the hardware components used in this project and their roles:

1. Webcam

A webcam plays a pivotal role in this project as it acts as the system's "eye." Its main function is to capture real-time facial images of individuals. These images are then processed by the software to identify whether the captured face matches any of the faces in the trained dataset of missing individuals.

Why a Webcam is Essential:

- **Real-Time Image Capture:** The project relies on real-time face detection. The webcam captures live video frames, from which faces are extracted and passed to the detection algorithm.
- **Input for Haar Cascade and YOLOv8:** Both Haar Cascade and YOLOv8 models need visual data as input. A webcam allows dynamic, live data to be processed on the fly.
- **Flexibility in Placement:** Webcams are easy to integrate into laptops or desktops and can also be positioned in public places like schools, railway stations, or shopping malls to monitor for missing children.
- **Multiple Frame Rates & Resolutions:** Modern webcams offer adjustable frame rates and resolutions, helping ensure high-quality image capture which is essential for precise facial recognition.

Recommended Features:

- Minimum resolution: 720p (for acceptable image clarity)
- Frame rate: 30 fps (for smooth detection)
- USB interface (for plug-and-play functionality)

2. Computer/Laptop (Minimum 4GB RAM and i5 Processor)

The core processing tasks in this project — such as face detection, image processing, YOLOv8 model inference, and running the Flask server — are executed on a computer or laptop. Thus, the specifications of the system directly influence the speed and accuracy of the detection process.

Why 4GB RAM is Required:

- **Efficient Memory Management:** Image processing and model inference involve handling large matrices and pixel data. A minimum of 4GB RAM ensures smooth operation of Python scripts and associated libraries.
- **Avoids Crashing:** Running heavy tasks like real-time YOLOv8 detection can be memory-intensive. With at least 4GB of RAM, the chances of system crashes or delays are minimized.

Why an i5 Processor (or equivalent) is Recommended:

- **Parallel Processing:** The Intel Core i5 processor comes with multiple cores and threads that support parallel execution, which is beneficial for real-time detection.
- **Model Execution Speed:** YOLOv8 and image augmentation algorithms are computationally heavy. An i5 processor provides the necessary power to process frames in real time without bottlenecks.
- **Flask Server Handling:** When combined with Flask for email integration and user interface, the CPU handles concurrent HTTP requests. An i5 processor ensures these requests are managed efficiently without lag.

System Usage Scenarios:

- **Model Training (Optional locally, but better on cloud):** For those who wish to train YOLOv8 on their local system, a more powerful GPU-based system is preferable. However, in this project, training is typically done beforehand using platforms like Google Colab, and only the .pt model is used locally.
- **Model Inference:** The trained YOLOv8 model runs on the local machine, taking webcam input and predicting matches in real time.
- **Email Notifications:** The system also processes logic for checking match confidence levels and triggers Flask-Mail to send alerts via email.

Additional Suggestions:

- If available, a system with a dedicated GPU (like NVIDIA GTX 1050 or higher) can accelerate detection.
- An SSD (Solid-State Drive) is preferred over an HDD for faster read/write operations, especially when dealing with image datasets.

3. Internet Connection

An active and stable internet connection is another key requirement for the proper functioning of the email alert feature using **Flask-Mail**. This feature ensures that when a missing person is detected by the system, an automated email is sent to the concerned authorities or family members.